# Convergence with Limited Visibility by Asynchronous Mobile Robots

Branislav Katreniak

Department of Computer Science, Faculty of Mathematics, Physics and Informatics,
Comenius University, Bratislava
katreniak@dcs.fmph.uniba.sk

**Abstract.** Consider a community of simple autonomous robots freely moving in the plane. The robots are decentralized, asynchronous, deterministic without the common coordination system, identities, direct communication, memory of the past, but with the ability to sense the positions of the other robots. This paper presents a distributed algorithm for the convergence problem with limited visibility in 1-bounded asynchrony. The presented algorithm also solves the convergence problem with unlimited visibility in full asynchrony without the need for the multiplicity detection.

## 1 Introduction

This paper deals with the study of the asynchronous distributed system of autonomous mobile robots called CORDA. The robots are anonymous, have no common knowledge, no common sense of the direction (e.g. compass), no central coordination and no means of direct communication. Behavior of these robots is quite simple. Each of them idles, observes, computes and moves in a cycle. In particular, each robot is capable of sensing the positions of the other robots with respect to its local coordination system, performing local computations on the observed data and moving toward the computed destination. The movement may stop before robot gets to the destination or the robot may not move at all. The robots' cycles and their phases are not synchronized and they can take arbitrary long time bounded only by a constant unknown to the algorithm. Local computation is done according to a deterministic algorithm that takes the sensed robots positions as the only input and returns the destination point toward which the executing robot moves. All robots perform the same algorithm.

The main research in this area is focused on understanding conditions necessary to complete given tasks, e.g. exploring the plane ([CP06]), forming particular patterns ([FPSW08], [SY99b], [Kat05], [DP07]), converge to a single point ([CP05]), gathering in a single point([CFPS03], [Pre05], [FPSW01], [Ka06]), flocking [YSDT09], etc. We are interested in algorithms with the correctness proof, not only justified by simulations.

The simplest studied problem in this model is the *convergence problem*: the robots must converge together to any point on the plane. This problem was solved in [CP05] with unlimited visibility in the full asynchrony. Every robot chooses as its destination the center of gravity of all observed robots positions. This algorithm is simple and correct. But it requires the *strong multiplicity*

*detection*: when more robots are at the same place, the robots sensors must tell how many robots share the place.

We found an alternative algorithm for this problem that does not require the multiplicity detection: every robot finds the furthest observed robot and moves halfway toward it. This algorithm is also very simple and we prove its correctness in this paper. And it does not require the multiplicity detection.

According to the CORDA model definition, our algorithm is better. It does not require the multiplicity detection. On the other side, we can argue that to implement the center of gravity algorithm, the sensors can be simpler and directly return the destination point. And it is possible to argue that our algorithm requires sensors to return only the position of the furthest robot.

The main focus is on the *convergence problem with limited visibility*: the robots only see those other robots that are within a fixed distance and they must converge together to any point on the plane, provided that the visibility graph is initially connected.

Paper [FPSW01] shows that the robots can converge with limited visibility in full asynchrony when they are equipped with the compass. Even more, they are able to *gather* at one point in finite time.

The convergence problem with limited visibility is much more difficult without the compass. Paper [SY99a] provides an algorithm and proves its correctness in pseudosynchronous settings, where the observation phases of all robots are globally synchronized[1]. Every pair of mutually visible robots maintains the mutual visibility. They restrict their destination to be inside the circle with the center at their middle and with the radius equal to half of the visibility distance. In pseudosynchronous settings, both robots will be again within the visibility distance in the next observation phase.

We aim to solve this problem with more asynchrony and we present an algorithm for the convergence problem with limited visibility in *$k$-bound asynchrony* with $k = 1$. Robots are asynchronous in that one robot may begin an activation cycle while another robot finishes one. We assume that the scheduler is $k$-bounded: from the moment one robot observes the current situation to the moment it finishes its movement, no other robot performs more than $k$ observations. Compared to definition in [YSDT09] where the k-bounded asynchrony was introduced, the robots are allowed to spend an arbitrary long time in their idle phase.

In this paper we prove, that the convergence problem with limited visibility is solvable in 1-bound asynchrony. In section 3 we present the restrictions on robots movements that ensure that the visibility graph stays connected. In section 4 we present an algorithm respecting these restrictions. In section 5 we prove that robots executing the proposed algorithm converge toward a point. In appendix we prove that the algorithm in [SY99a] is not correct in 1-bound asynchrony.

[1] The paper talks about asynchronous settings, but it provides only the simulation results.

## 2 Model and definitions

Each robot is viewed as a point in a plane equipped with sensors. It can observe the set of all points which are occupied by at least one other robot. Note that the robot only knows whether there are any other robots at a specific point, but it has no knowledge about their number (i.e. it cannot tell how many robots are at a given location). The *local view* of each robot consists of a local unit of length, an origin (w.l.o.g. the position of the robot in its current observation), an orientation of angles and the coordinates of the observed points. No kind of agreement on the unit of length, the origin and the orientation of angles is assumed among the robots. They are used only locally between observation, calculation and movement phases.

A robot is initially in the *waiting phase*. Asynchronously and independently from the other robots, it *observes* the environment (*Observe phase*) by activating its sensors. The sensors return a snapshot of the world, i.e. the set of all points occupied by at least one other robot, with respect to the local coordinate system. Then, based only on its local view of the world, the robot *calculates* its destination point (*Compute phase*) according to its deterministic algorithm (the same for all robots). After the computation the robot *moves* toward its destination point (*Move phase*); if the destination point is the current location, the robot stays on its place. The movement may stop before the robot reaches its destination (e.g. because of limits to the robot's motion energy). The robot then returns to the waiting state. The sequence *Wait – Observe – Compute – Move* forms a *cycle* of a robot.

To ensure progress, the model provides *progress condition*: Every robot moves at least a fixed distance $\theta$ once per a fixed number of cycles $Q$; constants $\theta$ and $Q$ are unknown for the algorithm.

The robots are *oblivious*: they do not remember any previous observations nor computations performed in any previous cycles.

The robots are *anonymous*: they are indistinguishable by their appearance, and they do not have any kind of identifiers that can be used during the computation.

The robots have *no means of direct communication*: any communication occurs in a totally implicit manner by observing the other robots positions.

The robot's sensors have *limited visibility*: robot observes only those other robots that are within a global fixed radius $r$.

The robots don't know $r$, but they can compute its lower bound as the distance as the distance to the furthest observed robot. We set, w.l.o.g., the local unit $1_R$ for robot $R$ as the distance to the furthest observed robot in the last observation.

The *full activation cycle* for any robot: the interval from the snapshot in the observation phase (included) to the next snapshot in the next observation phase (excluded).

The scheduler is *k-bounded*: from the moment one robot observes the current situation to the moment it finishes its movement, no other robot performs more than $k$ starts of the full activation cycles (snapshots).

We say that two robots $R_1$ and $R_2$ are initially connected by the *visibility edge*, when the initial distance $|R_1R_2| \leq r$. The *visibility graph* is the graph of the visibility edges.

*Convergence problem with limited visibility:* Find an algorithm for robots such that for any given group of robots in the plane with a the initial visibility graph being connected and any correct scheduler, the convex hull of all robots converges toward a point.

In further text $\mathcal{O}_V(b)$ denotes closed neighborhood of point $V$: $\mathcal{O}_V(b) = \left\{v \in \mathbb{R}^2; |Vv| \leq b\right\}$.

# 3 Connectivity Preservation with 1-bounded Asynchrony

If the initial distance of two robots $A$, $B$ is not more than the visibility distance $r$ (i.e. robots see each other), we put between them a visibility edge. The problem definition guarantees that the visibility graph is initially connected. If we preserve all visibility edges, the visibility graph stays connected indefinitely.

Robots connected with the visibility edge have to restrict their movements in order to preserve their mutual visibility. We are looking for such restrictions on robots movements that preserve the local visibility edges and that allow robots to converge together globally.

## 3.1 Connectivity Invariant

We introduce the *invariant*: if two robots $A$, $B$ are preserving the visibility, their destinations $A_T$, $B_T$ must always be inside $\frac{1}{2}$ radius from $C = \frac{A+B}{2}$.

The invariant initially holds for all initial visibility edges: $A_T = A$, $B_T = B$ and $|AB| \leq 1$. If the invariant holds indefinitely for any initial edge, the visibility graph stays connected indefinitely.

## 3.2 Invariant Preservation

Idea from article [SY99b]. Let $A$, $B$ be two robots at points $A_0$, $B_0$ preserving the visibility. Let $C = \frac{A_0+B_0}{2}$, $d = |A_0B_0|$. Refer to Figure 1. W.l.o.g. we set the coordinate system to origin in $C$; $C = [0,0]$. and the unit of distance to $r$; $r = 1$. Robots positions are $A_0 = [-\frac{d}{2}, 0]$, $B_0 = [\frac{d}{2}, 0]$.

When robot $B$ observes robot $A$, robot $B$ does not know the destination $A_T$ of robot $A$. Robot $B$ only knows from the invariant:

$$|CA_T| \leq \frac{1}{2}$$

$$A_T = [t\cos(\gamma), t\sin(\gamma)]; 0 \leq t \leq \frac{1}{2}; 0 \leq \gamma \leq 2\pi$$

Robot $A$ may finish its movement before it gets to $A_T$, but this case is covered by considering any $A_T$. Robot $B$ chooses its target at some point $B_T$; $B_T = [x, y]$ and starts the movement toward $B_T$. Robot $B$ may finish its movement before it gets to $B_T$, but this case is covered by considering any $B_T$.

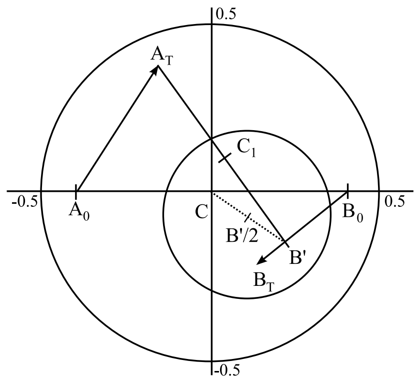



**Fig. 1.** Connectivity invariant

If robot $A$ does no observation during the movement of robot $B$, robot $A$ must be idling at end of the movement of robot $B$ (1-bound asynchrony). If we restrict $|CB_T| \le \frac{1}{2}$, both $A_T$ and $B_T$ are inside the circle at center $C$ and radius 1. Thus, both robots are idling within the distance 1 and the invariant holds.

If robot $A$ observes robot $B$ during the movement of robot $B$, it calculates new destination based on this observation. We have to ensure, that the invariant holds at the moment of the observation. Because of the 1-bound asynchrony, robot $A$ can perform the observation only once until robot $B$ finishes its movement to $B_T$. Refer to Figure 1. Robot $A$ observes robot $B$ at $B' = B_0 + \alpha(B_T - B_0); 0 \le \alpha \le 1$. To prove the invariant preservation, we have to show that the invariant now holds from robot's $A$ perspective: $C_1 = \frac{A_T + B'}{2}$, $|C_1 B_T| \le 1$.

$$
\begin{aligned}
&B = [\frac{d}{2}, 0];\ B_T = [x, y] \\
&0 \le \alpha \le 1;\ 0 \le d \le 1 \\
&B' = [x', y'] = B + \alpha(B_T - B) = \left[\frac{d}{2} + \alpha\left(x - \frac{d}{2}\right), \alpha y\right]
\end{aligned}
$$

We are looking for a condition for $B_T$ ensuring that the invariant holds for robot $A$ at the moment of the observation. Let's fix $B'$ to $B = [x, y]$ and watch possible positions of $C_1$ for different $A_T$. We want to find $A_T$ with maximal $|C_1 B_T|$.

$$
C1 = \frac{A_T + B'}{2} = \left[\frac{x'}{2} + \frac{t}{2}\cos(\gamma), \frac{y'}{2} + \frac{t}{2}\sin(\gamma)\right]; 0 \le t \le \frac{1}{2}; 0 \le \gamma \le 2\pi
$$

It means that points of $C_1$ belong to circle with center $\frac{B'}{2}$ and radius $\frac{1}{4}$. How far can be point inside this circle from $B_T$? It is $\left|B_T \frac{B'}{2}\right| + \frac{1}{4}$. Thus, if robot $B$ chooses such destination $B_T$ that $\left|B_T \frac{B'}{2}\right| \le \frac{1}{4}$, the invariant holds.

$$\left|B_T \frac{B'}{2}\right|^2 \leq \left(\frac{1}{4}\right)^2$$

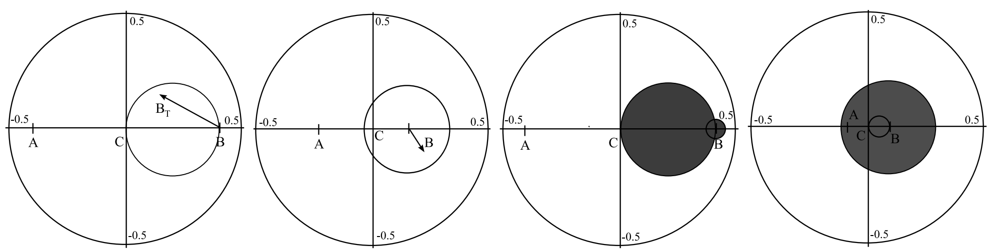
$$\left(x - \frac{d(1-\alpha)}{2(2-\alpha)}\right)^2 + y^2 \leq \left(\frac{1}{2(2-\alpha)}\right)^2 \tag{1}$$

Inequality 1 (calculated in appendix) together with inequality $|CB_T| \leq \frac{1}{2}$ are sufficient to preserve the invariant. We introduce two independent restrictions fulfilling the inequalities: *move toward* and *move around*. They provide two options how the robots can move in order to preserve the invariant.

### 3.3 Move toward

**Theorem 1.** *Let robot $B$ calculate its destination $B_T$; $|\frac{C+B}{2} B_T| \leq \frac{d}{4}$. The invariant is then preserved.*

*Proof.* Theorem 1 allows robot $B$ to move to any point inside the circle over diameter $CB$. Refer to Figure 2(a).




(a) Move toward (b) Move around (c) Combined move (d) Combined move

**Fig. 2.**

Inequality 2 expresses the circle over diameter $CB$. We need to prove that inequality 2 implies inequality 1. As both inequalities 2 and 1 specify a circle, we show that the first circle is inside the second one. As both centers are on the horizontal axis, it is sufficient to compare the leftmost and the rightmost points. Both comparisons easily hold.

$$\left(x - \frac{d}{4}\right)^2 + y^2 \leq \frac{d^2}{16} \tag{2}$$

$$\frac{d(1-\alpha)}{2(2-\alpha)} - \frac{1}{2(2-\alpha)} \leq \frac{d}{4} - \frac{d}{4}$$

$$\frac{d(1-\alpha)}{2(2-\alpha)} + \frac{1}{2(2-\alpha)} \geq \frac{d}{4} + \frac{d}{4}$$

□

### 3.4 Move around

**Theorem 2.** *Let robot $B$ calculate its destination $B_T$; $|BB_T| \leq l$; $l = \frac{1-d}{4}$. The invariant is then preserved.*

*Proof.* Theorem 2 allows robot $B$ to move to any point inside the circle with center $B$ and radius $l$ (circle is passing $\frac{B+[0.5,0]}{2}$). Refer to Figure 2(b).

Inequality 3 expresses the circle with center $B$ and radius $l$. We need to prove that inequality 3 implies inequality 1. Both equations again specify circles with the centers on the horizontal axis. We compare the leftmost and the rightmost points. Both comparisons easily hold.

$$l = \frac{1-d}{4}; \ \left(x - \frac{d}{2}\right)^2 + y^2 \leq l^2 \tag{3}$$

$$\frac{d(1-\alpha)}{2(2-\alpha)} - \frac{1}{2(2-\alpha)} \leq \frac{d}{2} - \frac{1-d}{4}$$
$$\frac{d(1-\alpha)}{2(2-\alpha)} + \frac{1}{2(2-\alpha)} \geq \frac{d}{2} + \frac{1-d}{4}$$

□

The calculation of $B_T$ for move around uses the visibility radius $r = 1$ unknown for the robot's algorithm. If robot $B$ uses the distance to the furthest observed robot $\mathit{1}_B$ instead of $r = 1$ for the calculation of $l$, the robot may be allowed to move less because $\mathit{1}_R \leq 1$, but the Theorem 2 holds.

## 4 Algorithm

When robot $R$ observes other robots, it receives a set of robots positions $\{R_1..R_n\}$ excluding $R$.

Robot $R$ cannot distinguish whether it maintains the visibility edge with particular robot $R_i$ or not. We restrict robot's movements to maintain the visibility with all observed robots. Every robot $R_i$ specifies the set of allowed target points $T_i$ as the union of allowed targets in *move toward* and in *move around*. Refer to figures 2(c), 2(d).

Let $S$ be the global convex hull of all robots at the time of the observation. Let $S_R$ be the convex hull of all robots observed by $R$ including $R$; $S_R = \{R_1..R_n\} \cup R$. From the global point of view, robots inside $S$ can move wherever they like, they just should not move toward the boundary of $S$. The robots at the boundary of $S$ should move inside $S$. Robot $R$ does not know the global $S$. But it knows the local subset $S_R$; $S_R \subseteq S$. We express the global goal as another set of allowed target points $T_S$. Let $T_S$ be the set of points that are not further than halfway toward the boundary of $S_R$: $T_S = \{v \in \mathbb{R}^2; (R + 2(v - R)) \in S_R\}$.

We have the list of sets with allowed target points and all of them must apply. Let $T$ be their intersection: $T = \bigcap_{i=1}^{n} T_i \cap T_S$.

Robot $R$ chooses as its destination $R_T$ the point in $T$ that is furthest from $R$. If more than one point apply, it chooses point with the smallest local angle. The robot simply moves that direction, where it is allowed to move furthest.

### 4.1 Algorithm with Unlimited Visibility

When the robots have the unlimited visibility, they don't need to maintain the visibility edges. They see each other all the time. The set of allowed target points $T$ becomes $T_S$ and the algorithm becomes simple: Robot $R$ finds the furthest observed robot $R_m$ and moves halfway toward it: $\frac{R+R_m}{2}$.

## 5 Convergence

We are going to prove that the constructed algorithm is correct and solves the convergence problem with limited visibility. We have to prove that the robots converge toward a point for any initial configuration and for any scheduler.

The model is asynchronous. But with the fixed initial situation and with the fixed scheduler, the whole sequence of robot's activations, observations and movements is fixed. We are interested in those time instants, when one or more robots performed the observation snapshot. We mark the initial time as $t_0$ and the times of the observations as $t_1, t_2, \ldots$

When a robot calculates its new destination, the result of the calculation is a pure function of the observed input. We hide the calculation's duration to the asynchrony of the movement phase and we say that the destination is calculated and applied immediately at the time of the observation.

Let $S(t)$ be the convex hull of all robots positions and of all robots destinations at time $t$. Let $S_R(t)$ be the convex hull of robots positions observed by robot $R$ at time $t$ including robot $R$; $S_R(t) \subseteq S(t)$. The value $S_R(t)$ is defined only when robot $R$ performs an observation at time $t$. We are going to prove that $S(t)$ converges toward a point.

**Theorem 3.** *Convex hull $S(t)$ never grows: $\forall i; S(t_i) \supseteq S(t_{i+1})$*

*Proof.* Consider any robot at position $R$ with destination $R_T$. Because both points $R$ and $R_T$ are in $S(t)$, also the whole line segment $RR_T$ is in $S(t)$. The robot's movement cannot enlarge $S(t)$. The new destination for robot $R$ is calculated inside $S_R(t); S_R(t) \subseteq S(t)$. □

Since convex hull $S(t)$ never grows, it converges to some shape.

**Theorem 4.** *Convex hull $S(t)$ converges uniformly toward a convex polygon $S$ of at most $2n$ points.*

*Proof.* We know that $S(t+1) \subseteq S(t)$ and that every convex hull $S(t)$ is a closed convex shape. Then $S = \bigcap_{t=0}^{\infty} S(t)$ is again a closed convex shape.

*Uniform convergence*: $\forall \varepsilon > 0;\ \exists t_1;\ \forall t_2 \geq t_1;\ S(t_2) \subseteq S + \varepsilon$. By contradiction, let every $S(t)$ contain point $w_t \notin S + \varepsilon$. Sequence $w_t$ is an infinite sequence of points bounded by an initial convex hull $S(0)$ and must contain an infinite subsequence of points converging toward a point $w$; $w \notin S + \varepsilon$. But $\forall t; w \in S(t)$; $S(t)$ is closed; $S(t+1) \subseteq S(t)$, thus $w \in S$. Contradiction.

*$S$ is polygon of at most $2n$ points*: Every $S(t)$ is the convex hull of at most $2n$ points, thus it is a polygon of at most $2n$ points. We choose a point $V$ inside $S$.

We define function $f_t(\alpha)$ as the distance from $V$ to the intersection of $S(t)$ with ray from $V$ with angle $\alpha$. Polygon $S(t)$ of at most $2n$ points maps to polyline $f_t$ of at most $2n$ lines. As sequence $S(t)$ is converging uniformly, sequence of $f_t$ is uniformly converging too. Thus $f_t$ converges uniformly to a polyline of at most $2n$ lines and also $S$ is polygon of at most $2n$ lines. $\square$

Consider an angle $\alpha$; $\alpha < \pi$ based at a vertex $V$. Suppose that all robots are inside $\alpha$ and stay inside $\alpha$ forever. This corresponds to the situation at any vertex of any convex hull $S(t)$. Consider a robot $R$ "close" to the vertex $V$. The following theorem says that robot $R$ moves "away" of $V$ and stays "away" of $V$. Since the robot $R$ has no idea about any global unit of distance, it measures the distances "close" and "away" in its local unit $1_R$ set to the distance of the furthest observed robot.

**Theorem 5.** *Let all robot positions and destinations be inside an angle at vertex $V$ of size $\alpha$; $\alpha < \pi$. We find value $c$; $0 < c \leq \frac{1}{12}$ such that no robot $R$ ever chooses its destination inside $\mathcal{O}_V(c1_R)$*

*Proof.* We set the unit of distance for this proof to $1 = 1_R$, i.e. the distance to the furthest robot observed by $R$.

Let $e = \frac{1}{24}\cos\frac{\alpha}{2}$. Suppose that robot $R$ is inside $\mathcal{O}_V(e)$ (i.e. $|VR| \leq e$), it performs an observation and calculates the destination $R_T$. We analyze the lower bound for $|RR_T|$.

Let $p$ be the ray starting at $R$ and parallel with axis of $\alpha$. Refer to Figure 3(a). We analyze how far robot $R$ can move in direction $p$.

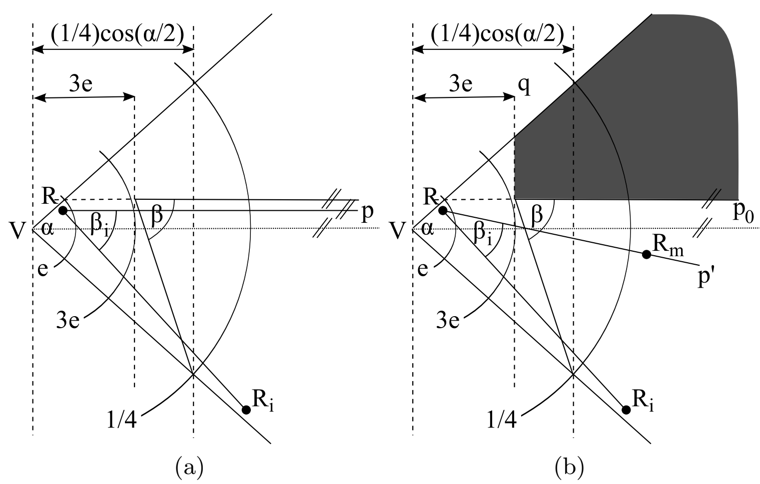



**Fig. 3.** Setup in Theorem 5

Let $R_i$ be any robot observed by $R$.

- If $|VR_i| \leq \frac{1}{4}$, then $|RR_i| \leq |VR_i| + |VR| \leq \frac{1}{4} + e \leq \frac{1}{3}$. Robot $R_i$ allows robot $R$ to *move around* in direction $p$ at least $\frac{1}{6}$.

– If $|VR_i| > \frac{1}{4}$, then $|RR_i| \geq |VR_i| - |VR| \geq \frac{1}{4} - e \geq \frac{1}{6}$. Let $\beta_i$ be the angle between $p$ and $RR_i$. Robot $R_i$ allows $R$ to *move toward* in direction $p$ at least $\frac{1}{2}|RRi|\cos\beta_i \geq \frac{1}{12}\cos\beta_i$.

Let $\beta$; $\beta < \frac{\pi}{2}$ (calculated in appendix) be an upper bound for any $\beta_i$; $\beta_i \leq \beta$ and also for $\frac{\alpha}{2}$; $\frac{\alpha}{2} \leq \beta$. Robot $R_i$ allows robot $R$ to move in direction $p$ either $\frac{1}{6}$ or $\frac{1}{12}\cos\beta_i$. We mark this allowed distance as $f_i$; $f_i = min(\frac{1}{6}, \frac{1}{12}\cos\beta_i) = \frac{1}{12}\cos\beta_i \geq \frac{1}{12}\cos\beta \geq \frac{1}{24}\cos\beta$. We mark the lower bound for $f_i$ as $f$; $f = \frac{1}{24}\cos\beta$. Any observed robot allows robot $R$ to move in the direction $p$ at least the distance $f$. Note that $f = \frac{1}{24}\cos\beta \leq \frac{1}{24}\cos\frac{\alpha}{2} = e$.

If the restriction to move only halfway toward the local convex hull $S_R(t)$ allows $R$ to move in the direction $p$ at least the distance $f$, we have the lower for $|RR_T|$; $|RR_T| \geq f$.

Otherwise refer to Figure 3(b). Let $R_m$ be a robot at distance 1. W.l.o.g, let $R_m$ be bellow the axis of $\alpha$. Construct line $q$ perpendicular to the axis of $\alpha$ at distance $3e$ from $V$. Construct line $p_0$ parallel to the axis of $\alpha$ at distance $e\sin\frac{\alpha}{2}$ (upper bound for $p$). Because the restriction to move only halfway toward the local convex hull $S_R(t)$ applies, the points at $p$ further than $2f$ from $R$ are not in $S_R(t)$. Thus there is no robot at both sides of $p$ further than $2f < 2e$. Thus there is no robot behind $q$ on the upper side of $p_0$ (figure's gray area). Let $p'$ be the ray from $R$ to $R_m$. We analyze how far $R$ can move in direction $p'$.

Let $R_i$ be any robot observed by $R$.

– If $|VR_i| \leq \frac{1}{4}$, then $|RR_i| \leq |VR_i| + |VR| \leq \frac{1}{4} + e \leq \frac{1}{3}$. Robot $R_i$ allows robot $R$ to *move around* in direction $p'$ at least $\frac{1}{6}$.
– If $|VR_i| > \frac{1}{4}$, then $|RR_i| \geq |VR_i| - |VR| \geq \frac{1}{4} - e \geq \frac{1}{6}$. Let $\beta'_i$ be the angle between $p'$ and $RR_i$. Robot $R_i$ allows $R$ to *move toward* in direction $p'$ at least $\frac{1}{2}|RRi|\cos\beta'_i \geq \frac{1}{12}\cos\beta'_i$.

The angle $\beta$ is again an upper bound for any $\beta'_i$; $\beta'_i \leq \beta$. Thus, any observed robot allows robot $R$ to move in the direction $p'$ again at least the distance $f$. The convex hull restriction does not apply in the direction $p'$, because there is robot $R_m$ on ray $p$ at distance 1.

We proved that when $|VR| \leq e$, then robot's $R$ destination is at least at the distance $f$. We set the value $c$ asked in the theorem to $c = \frac{1}{4}f\cos\frac{\alpha}{2}$. Note that $c \leq \frac{f}{4} < f \leq e$.

– If $|VR| \leq c$, robot chooses its destination outside of $\mathcal{O}_V(f - c) \supseteq \mathcal{O}_V(f(1 - \frac{1}{4}\cos\frac{\alpha}{2})) \supseteq \mathcal{O}_V(\frac{3f}{4}) \supseteq \mathcal{O}_V(\frac{3c}{\cos(\alpha/2)}) \supseteq \mathcal{O}_V(c)$.
– If $|VR| \leq \frac{f}{2}$, $R$ moves at least $f$ and its destination will be outside $\mathcal{O}_V(\frac{f}{2}) \supset \mathcal{O}_V(c)$.
– If $|VR| > \frac{f}{2}$, $R$ moves at most $|VR|$ toward $V$ and its destination cannot be inside $\mathcal{O}_V(\frac{f}{4}) \supseteq \mathcal{O}_V(c)$.

□

From the Theorem 4 we know that $S$ is a convex polyline of at most $2n$ lines. For every $\varepsilon > 0$ exists such $t$ that $S \subseteq S(t) \subseteq S + \varepsilon$. Consider any vertex $A$ of $S$ with an angle $\alpha$ and with an axis $p$. Angle $\alpha$ is an angle in the convex

hull, thus $\alpha < \pi$. Refer to Figure 4. Construct line $q$ perpendicular to $p$ at $A$. Points on the convex hull's side of $q$ are *before* $q$, points on the other side of $q$ are *behind* $q$.

**Theorem 6.** *For every $b > 0$ exists $\varepsilon$; $0 < \varepsilon \leq \frac{b}{12}$ such that if any robot $R_1$ observes robot $R_2 \notin \mathcal{O}_A(b)$, $R_1$ chooses its destination before $q$.*

*Proof.* Consider an upper bound $\varepsilon_1$ for $\varepsilon$; $\varepsilon \leq \varepsilon_1 = \frac{b}{12}$. If $R_1$ behind $q$ observes $R_2 \notin \mathcal{O}_A(b)$; $1_{R_1} \geq |R_1R_2| \geq b - \varepsilon \geq \frac{11b}{12}$. Refer to Figure 4. We use Theorem 5 with vertex $V$ on $p$ behind $q$ at distance $|VA| = \frac{\varepsilon}{\sin(\alpha/2)}$, bounding angle parallel to $\alpha$, robot $R = R_1$. We get $c$.

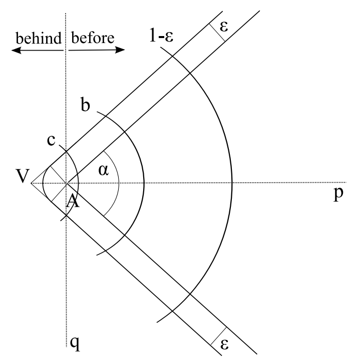



**Fig. 4.** Setup in Theorem 7

Let $\varepsilon = \min(c \cos\frac{\alpha}{2}\sin\frac{\alpha}{2}, \varepsilon_1)$; $\varepsilon \leq \varepsilon_1$; $\varepsilon \leq c\cos\frac{\alpha}{2}\sin\frac{\alpha}{2}$. If robot $R_1$ is behind $q$, $|VR_1| \leq \frac{|VA|}{\cos(\alpha/2)} = \frac{\varepsilon}{\cos(\alpha/2)\sin(\alpha/2)} \leq c$, $R_1 \in \mathcal{O}_V(c)$. From Theorem 5 we get that robot $R_1$ chooses its destination outside of $\mathcal{O}_V(c)$ and before $q$. $\square$

Now we are ready to prove the final Theorem.

**Theorem 7.** *Convex hull $S(t)$ converges toward point.*

*Proof.* By contradiction. Suppose that $S(t)$ converges toward a shape $S$ that is not a point. We are going to find such $\varepsilon$ that all robots with destinations behind $q$ move before $q$ in finite time. This will be the contradiction with the assumption that robots converge toward non-point $S$.

Consider an upper bound $b_1$ for $b$; $0 < b \leq b_1 = \frac{1}{12}$. We use Theorem 6 with declared $b$ and get $\varepsilon$; $\varepsilon \leq \frac{b}{12} \leq \frac{1}{144}$. We let the robots execute till $S \subseteq S(t) \subseteq S + \varepsilon$.

Refer to Figure 4. While a robot $R_1$ behind $q$ observes a robot $R_2 \notin \mathcal{O}_A(b)$, $R_1$ moves before $q$ in finite time and stays before $q$.

If a robot $R_1$ behind $q$ does not observe a robot $R_2 \notin \mathcal{O}_A(b)$, there is no robot in $\mathcal{O}_{R_1}(1)$ and thus no robot in $\mathcal{O}_A(1-\varepsilon)$. We know that the visibility graph is connected. Either all robots are in $\mathcal{O}_A(b)$ or there must be a robot $R_3$ in $\mathcal{O}_A(b)$ that preserves mutual visibility with a robot $R_4 \notin \mathcal{O}_A(1-\varepsilon)$.

Suppose the existence of $R_4 \notin \mathcal{O}_A(1-\varepsilon)$. Robot $R_3$ sees $R_4$, $1_{R_3} \geq 1-b-\varepsilon \geq \frac{10}{12}$. We use Theorem 5 with robot $R = R_3$, $V$ and $\alpha$ as in Theorem 6, we get $c$. Let $d_S$ be the diameter of $S$. Let $b = \min(\frac{11c}{12}, \frac{d_S}{2}, b_1)$; $b \leq \frac{d_S}{2}$; $b \leq b_1$.

If robot $R_4$ stays out of $\mathcal{O}_A(1-\varepsilon)$, robot $R_3$ moves in finite time (progress condition) out of $\mathcal{O}_A(b)$ and stays there. Robot $R_3$ observed all robots behind $q$ and maintains with them visibility. Robots behind $q$ see $R_3 \notin \mathcal{O}_A(b)$, move before $q$ in finite time (progress condition) and stay there.

If robot $R_4$ ever moves into $\mathcal{O}_A(1-\varepsilon)$, it observes robots behind $q$ and maintains with them visibility.

If robot $R_4$ stays out of $\mathcal{O}_A(b)$, robots behind $q$ see $R_4 \notin \mathcal{O}_A(b)$, move before $q$ (progress condition) in finite time and stay there.

If robot $R_4$ ever moves into $\mathcal{O}_A(b)$, we have at least one more robot in $\mathcal{O}_A(b)$. We repeat this proof again from the point where $R_1$ behind $q$ observes or does not observe some robot out of $\mathcal{O}_A(b)$. Either all robots behind $q$ get and stay before $q$ or all robots get into $\mathcal{O}_A(b)$. But all robots cannot be in $\mathcal{O}_A(b)$, it is contradiction with $b \leq \frac{d_S}{2}$. □

The convergence proof never uses restrictions imposed by the $k$-bound scheduler. It uses only the progress condition present also in the basic asynchronous model.

## 6 Conclusions and open questions

We proposed an algorithm for the convergence problem with limited visibility and we proved that it is correct when the asynchrony is limited to 1. Compared to algorithm in [SY99a], our algorithm works in more asynchronous model.

We concluded that the proposed algorithm solves the convergence problem also in asynchronous settings with unlimited visibility with no need for the multiplicity detection. Compared to the algorithm in [CP05], our algorithm does not require the multiplicity detection.

The convergence problem with limited visibility is open with asynchrony limited to a general constant $a$ and in asynchronous settings.

## References


[CFPS03] M. Cieliebak, P. Flocchini, G. Prencipe, and N. Santoro. Solving the gathering problem. In *30th International Colloquium on Automata, Languages and Programming (ICALP '03)*, pages 1181–1196. Springer Verlag, Berlin, 2003.

[CP05] Reuven Cohen and David Peleg. Convergence properties of the gravitational algorithm in asynchronous robot systems. *SIAM J. Comput.*, 34(6):1516–1528, 2005.

[CP06] R. Cohen and D. Peleg. Local algorithms for autonomous robot systems. In *13th International Colloquium On Structural Information And Communication Complexity (SIROCCO 13)*, pages 29–43, 2006.

[DP07] Y. Dieudonne and F. Petit. Swing words to make circle formation quiescent. In *14th International Colloquium On Structural Information And Communication Complexity (SIROCCO 14)*, pages 166–179. Springer Verlag, Berlin, 2007.

[FPSW01] P. Flocchini, G. Prencipe, N. Santoro, and P. Widmayer. Gathering of asynchronous mobile robots with limited visibility. In *18th Symposium on Theoretical Aspects of Computer Science (STACS 2001)*, pages 247–258. Springer Verlag, Berlin, 2001.

[FPSW08] Paola Flocchini, Giuseppe Prencipe, Nicola Santoro, and Peter Widmayer. Arbitrary pattern formation by asynchronous, anonymous, oblivious robots. *Theor. Comput. Sci.*, 407:412–447, November 2008.
[Ka06] B. Katreniak and J. Katreniaková. On the inability of gathering by asynchronous mobile robots with initial movements. In *ECAI 2006*, pages 255–259. IOS Press, Amsterdam, 2006.
[Kat05] B. Katreniak. Biangular circle formation by asynchronous mobile robots. In *12th International Colloquium On Structural Information And Communication Complexity (SIROCCO 12)*, pages 185–199. Springer Verlag, Berlin, 2005.
[Pre05] G. Prencipe. On the feasibility of gathering by autonomous mobile robots. In *12th International Colloquium On Structural Information And Communication Complexity (SIROCCO 12)*, pages 246–261. Springer Verlag, Berlin, 2005.
[SY99a] I. Suzuki and M. Yamashita. Distributed memoryless point convergence algorithm for mobile robots with limited visibility. *IEEE Transactions on Robotics and Automation*, 28(4):1347–1363, 1999.
[SY99b] Ichiro Suzuki and Masafumi Yamashita. Distributed anonymous mobile robots: Formation of geometric patterns. *SIAM Journal on Computing*, 28(4):1347–1363, 1999.
[YSDT09] Y. Yang, S. Souissi, X. Défago, and M. Takizawa. Fault-tolerant flocking of mobile robots with whole formation rotation. In *The IEEE 23rd International Conference on Advanced Information Networking and Applications, AINA 2009, Bradford, United Kingdom, May 26-29, 2009*, pages 830–837. IEEE Computer Society, 2009.

## A Algorithm in [SY99a] does not work in 1-bound asynchrony

Paper [SY99a] provides simulation results for asynchronous model. The correctness proof is done when: The time duration needed to complete a phase is negligibly small. We are going to prove that the algorithm presented in this paper does not work in 1-bound asynchrony.

Every pair of mutually visible robots maintains the mutual visibility. To preserve the visibility, robots restrict their destination to be inside the circle with the center at their middle and with the radius equal to half of the visibility distance. In pseudosynchronous settings, both robots will be again within the visibility distance in the next observation phase. This does not apply in 1-bound asynchrony.

Algorithm: Robots choose as their destination the center of the smallest enclosed circle of all observed robots. But they move only such a distance that they satisfy the visibility maintaining restriction.

We present scheduling for the algorithm proving that the algorithm is not correct in 1-bound asynchrony. Refer to Figure 5. The unit of distance is set to be equal to the visibility distance. In the initial configuration, all robots are in 1D space on the horizontal axis as the positions: $A = -0.25$, $B = 0.25$, $C = -1.25$, $D = 1.25$, $E = -2.25$. Scheduler activation sequence:

- A at -0.25 observes and sets target to $\frac{B+C}{2} = -0.5$
- B at -0.25 observes and sets target to $\frac{A+D}{2} = 0.5$
- A moves to -0.4 and is paused
- C at -1.25 observes and sets target to $\frac{A+E}{2} = -1.325$
- C moves to -1.325
- A finishes its movement to -0.5

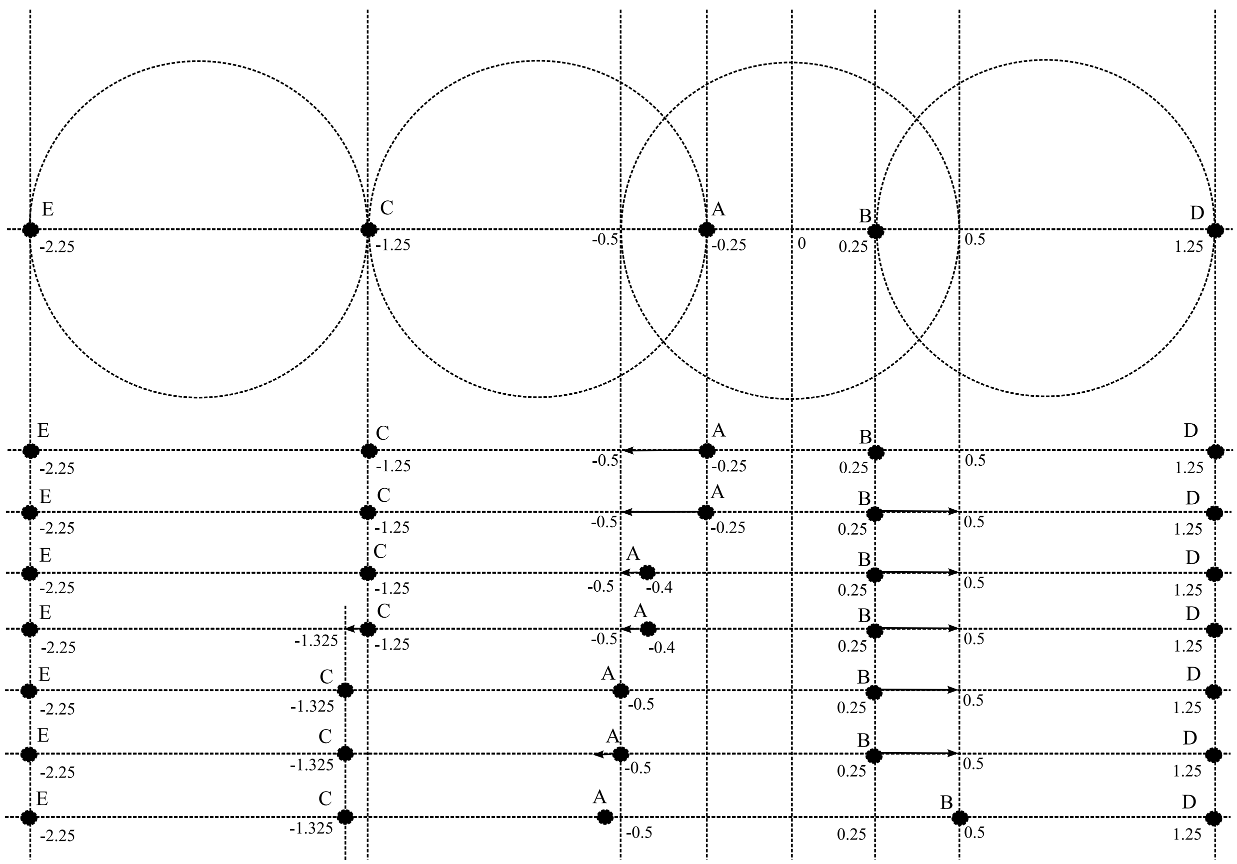



**Fig. 5.** Connectivity is broken

– A at -0.5 observes and sets target to $\frac{B+C}{2} = -0.5375$
– B moves to 0.5, A moves to -0.5375

Robots $D$ and $E$ are either idling or their movement is scheduled to be negligibly small. At the end, robot $A$ and robot $B$ broke their mutual visibility.

## B Calculation of equation 1

$$\begin{aligned}
\left| B_T \frac{B'}{2} \right|^2 &\leq \left(\frac{1}{4}\right)^2 \\
\left(x - \frac{\frac{d}{2} + \alpha\left(x - \frac{d}{2}\right)}{2}\right)^2 + \left(y - \frac{\alpha y}{2}\right)^2 &\leq \left(\frac{1}{4}\right)^2 \\
\left(\frac{2x - \frac{d}{2} - \alpha x + \frac{\alpha d}{2}}{2}\right)^2 + \left(\frac{2y - \alpha y}{2}\right)^2 &\leq \left(\frac{1}{4}\right)^2 \\
\left(\frac{x(2-\alpha) - \frac{d(1-\alpha)}{2}}{2}\right)^2 + \left(\frac{y(2-\alpha)}{2}\right)^2 &\leq \left(\frac{1}{4}\right)^2 \\
\left(x - \frac{d(1-\alpha)}{2(2-\alpha)}\right)^2 + y^2 &\leq \left(\frac{1}{2(2-\alpha)}\right)^2 \qquad (4)
\end{aligned}$$

## C Calculation of $\beta$ in theorem 5

$$\beta = \arctan \frac{\frac{1}{4}\sin\frac{\alpha}{2} + e\sin\frac{\alpha}{2}}{\frac{1}{4}\cos\frac{\alpha}{2} - 3e}$$

We have to prove that $\beta < \frac{\pi}{2}$. We show that value in arctan is positive.

$$\begin{aligned}
\frac{1}{4}\cos\frac{\alpha}{2} - 3e &> 0 \\
\frac{1}{4}\cos\frac{\alpha}{2} &> 3e \\
\frac{1}{4}\cos\frac{\alpha}{2} &> 3\frac{1}{24}\cos\frac{\alpha}{2} \\
\frac{1}{4} &> \frac{1}{8}
\end{aligned}$$

We also have to prove that $\beta \geq \frac{\alpha}{2}$

$$
\begin{aligned}
\beta &= \arctan \frac{\frac{1}{4}\sin\frac{\alpha}{2} + e\sin\frac{\alpha}{2}}{\frac{1}{4}\cos\frac{\alpha}{2} - 3e} \\
&\geq \arctan \frac{\frac{1}{4}\sin\frac{\alpha}{2}}{\frac{1}{4}\cos\frac{\alpha}{2}} \\
&= \arctan \frac{\sin\frac{\alpha}{2}}{\cos\frac{\alpha}{2}} \\
&= \frac{\alpha}{2}
\end{aligned}
$$